Police-Reported Crime Trends in West Yorkshire

June 30, 2025

Abdulquadri Abdulraheem

1 Introduction and Data Overview

1.1 Objective

- Analyse police-reported crime trends in West Yorkshire (April–September 2020).
- Quantify monthly crime counts, examine patterns by crime type and highlight data issues/unexpected features.

1.2 Data Source and Structure

The dataset consists of monthly CSV files (April–September 2020), originally from data.police.uk. Each file contains street-level crime across West Yorkshire.
- All files were programmatically loaded and combined for scalable, reproducible analysis:

[333]:

```
import pandas as pd
import os
base_path = r'C:\Users\PC\Desktop\LIDA task\crime_data\data'
months = ['2020-04', '2020-05', '2020-06', '2020-07', '2020-08', '2020-09']
crime_data = pd.DataFrame()
for month in months:
    file_name = f'{month}-west-yorkshire-street.csv'
    file_path = os.path.join(base_path, month, file_name)
    df = pd.read_csv(file_path)
    df['Month'] = month
    crime_data = pd.concat([crime_data, df], ignore_index=True)
```

1.3 Initial Data Exploration

The combined dataset contains **158,898 rows** and **13 columns**
- **Notable missing data:**
- `Crime ID` (29,689, ~19%)
- `Last outcome category` (31,334, ~20%)
- `Context` (158,898, 100%, always empty)
- **Unique crime types:** 15; Most frequent: *Violence and sexual offences* (57,483 records)
- **Most common location label:** "No Location" (3,488 records)

2 Data Cleaning and Preprocessing

- **Dropped irrelevant columns**: 9 columns remaining.
- **Removed records with missing crime type**: 2,000 dropped, remaining 156898 rows.
- **Filtered unrealistic geographic coordinates**: 3,637 dropped, remaining 153261 rows.
- **Removed duplicates**: 8,003 duplicates removed, remaining 145258 rows.
- **Standardised/filtered crime types:** 110 *Exclusive* dropped, retained official categories.
- **Final dataset:** 145,148 rows (from 158,898 original)
- **Official crime types remaining:** 14
- Annotated Cleaning Code:

[335]:

```python
# Always start with a fresh copy of the original data
crime_data_cleaned = crime_data.copy()
# Drop irrelevant columns
crime_data_cleaned = crime_data_cleaned.drop(columns=['Unnamed: 0', 'Context', 'Reported by', 'Falls within'])
# Remove rows with missing crime type
crime_data_cleaned = crime_data_cleaned.dropna(subset=['Crime type'])
# Filter out records with coordinates outside West Yorkshire
crime_data_cleaned = crime_data_cleaned[crime_data_cleaned['Longitude'].between(-2.5, -1.0) & crime_data_cleaned['Latitude'].between(53.0, 54.0)]
# Remove duplicate events
crime_data_cleaned = crime_data_cleaned.drop_duplicates(subset=['Crime ID', 'Longitude', 'Latitude', 'Month'])
# Standardize crime type & filter for official categories
crime_data_cleaned['Crime type'] = (crime_data_cleaned['Crime type'].astype(str).str.strip().str.title())
official_crime_types = ['Anti-Social Behaviour', 'Bicycle Theft', 'Burglary', 'Criminal Damage And Arson', 'Drugs', 'Other Crime', 'Other Theft', 'Possession Of Weapons', 'Public Order', 'Robbery', 'Shoplifting', 'Theft From The Person', 'Vehicle Crime', 'Violence And Sexual Offences']
crime_data_cleaned = crime_data_cleaned[crime_data_cleaned['Crime type'].isin(official_crime_types)]
# Standardize outcome, fill missing
crime_data_cleaned['Last outcome category'] = (crime_data_cleaned['Last outcome category'].fillna('Outcome Unknown').str.strip().str.title())
# Reset index
crime_data_cleaned.reset_index(drop=True, inplace=True)
print("Rows remaining after all cleaning steps:", len(crime_data_cleaned))
```

```
Rows remaining after all cleaning steps: 145148
```

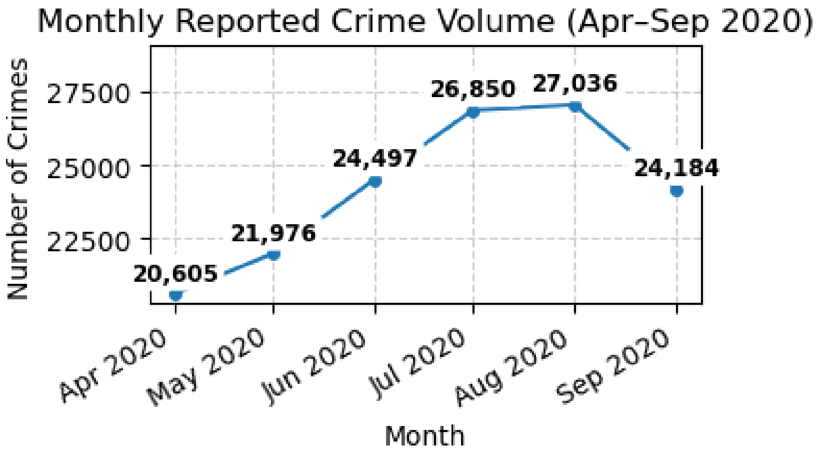
3 Exploratory Analysis and Crime Trends

3.1 Monthly Crime Trends

Crime increased from April (lowest) to August (peak), with a decline in September. This likely reflects the easing of lockdown restrictions, increased social activity and seasonal factors.

[522]:

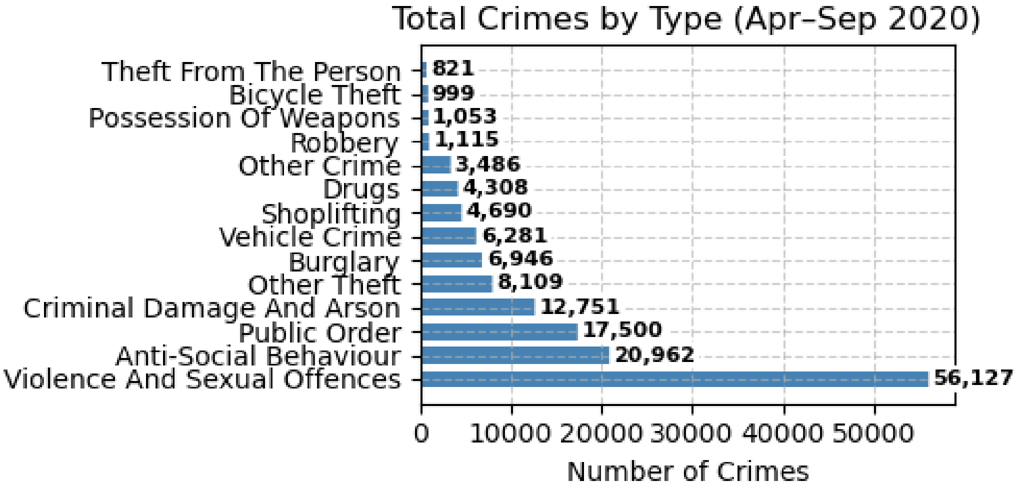
```
import matplotlib.pyplot as plt
import matplotlib.dates as mdates
# Ensure 'Month' is a datetime BEFORE grouping
crime_data_cleaned['Month'] = pd.to_datetime(crime_data_cleaned['Month'],␣
 ↪format='%Y-%m')
# Group by month
monthly_counts = crime_data_cleaned.groupby('Month').size()
plt.figure(figsize=(4, 2.6))
plt.plot(monthly_counts.index, monthly_counts.values, marker='o', markersize=4,␣
 ↪linewidth=1.5)
# Annotate each point with just the count
for x, y in zip(monthly_counts.index, monthly_counts.values): plt.text( x, y +␣
 ↪300, f"{y:,}", ha='center', va='bottom', fontsize=8.5, fontweight='bold',␣
 ↪color='black', bbox=dict(facecolor='white', edgecolor='none',␣
 ↪boxstyle='round,pad=0.25'))
plt.ylim(top=monthly_counts.max() + 2000)
plt.title("Monthly Reported Crime Volume (Apr-Sep 2020)")
plt.xlabel("Month")
plt.ylabel("Number of Crimes")
plt.grid(True, linestyle='--', alpha=0.6)
plt.gca().xaxis.set_major_formatter(mdates.DateFormatter('%b %Y'))
plt.gcf().autofmt_xdate()
plt.tight_layout()
plt.show()
```



3.2 Crime Types Distribution

Violence and sexual offences accounted for the largest, followed by anti-social behaviour and public order offences. This suggests persistent issues in violence and social disorder.

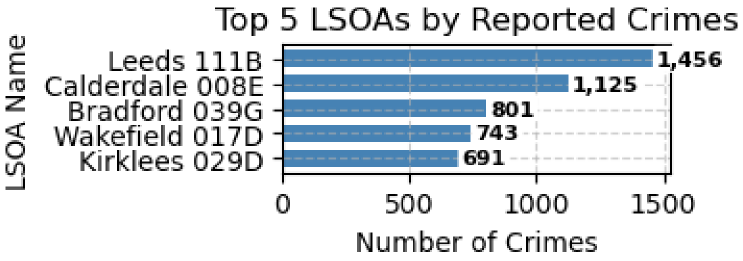
[532]:
```
crime_counts = crime_data_cleaned['Crime type'].value_counts()
plt.figure(figsize=(5.5, 2.8))
plt.barh(crime_counts.index, crime_counts.values, color='steelblue', height=0.7)
plt.title("Total Crimes by Type (Apr-Sep 2020)")
plt.xlabel("Number of Crimes")
plt.grid(True, linestyle='--', alpha=0.6)
# Add value labels to each bar
for i, v in enumerate(crime_counts.values):plt.text(v + 300, i, f"{v:,}",␣
 ↪va='center', fontsize=8, fontweight='bold', color='black',␣
 ↪bbox=dict(facecolor='white', edgecolor='none', boxstyle='round,pad=0.18'))
plt.tight_layout()
plt.show()
```



3.3 Top LSOAs by Crime (Area Hotspots)

The most affected LSOAs are in Leeds and Bradford, likely reflecting population density and urban dynamics. This may guide future resource allocation.

[541]:
```
top_lsoas = crime_data_cleaned['LSOA name'].value_counts().head(5)
plt.figure(figsize=(4, 1.6))
ax = plt.barh(top_lsoas.index, top_lsoas.values, color='steelblue', height=0.7)
plt.title("Top 5 LSOAs by Reported Crimes")
plt.xlabel("Number of Crimes")
plt.ylabel("LSOA Name")
plt.grid(True, linestyle='--', alpha=0.6)
plt.gca().invert_yaxis()
# Label each bar with just the count
for i, v in enumerate(top_lsoas.values): plt.text(v + 15, i, f"{v:,}",␣
 ↪va='center', fontsize=8, fontweight='bold', color='black',␣
 ↪bbox=dict(facecolor='white', edgecolor='none', boxstyle='round,pad=0.18'))
plt.tight_layout()
plt.show()
```



4 Data Limitations and Recommendations

4.1 Data Quality and Limitations

- 14.4% of records lack a unique `Crime ID`, mostly affecting “Anti-Social Behaviour” cases.
- All key analysis fields are complete after cleaning.
- Location-based analysis may be skewed by generic or repetitive location names.
- Only reported crimes are included; underreporting is possible.
- No supporting context fields or timestamps, limiting temporal and contextual analysis.

4.2 Recommendations and Future Work

Given the time and space constraints of this exercise, the analysis focused on the most critical trends. The following steps are recommended for further insights and impact:

4.2.1 Immediate Next Steps

- **Crime Type Trends:** Analyse month-by-month changes for each crime type.
- **Location Detail:** Identify the top 10 LSOAs and key locations with the highest crime rates.
- **Outcome Analysis:** Assess case outcomes by crime type and area to spot resolution issues.
- **Data Quality Check:** Address any missing or inconsistent data and vague locations.

4.2.2 Long-Term Opportunities

- **Hotspot Mapping:** Use geospatial heatmaps to pinpoint crime hotspots.
- **Finer Temporal Trends:** Analyse data weekly or daily if available.
- **Predictive Modelling:** Build models to forecast future crime or evaluate interventions.
- **Data Quality Improvement:** Collaborate with providers to enhance data consistency.